Data Communication and Net-Centric Computing

COSC 1111/2061/1110

Lecture 12
Emerging Networking Technologies

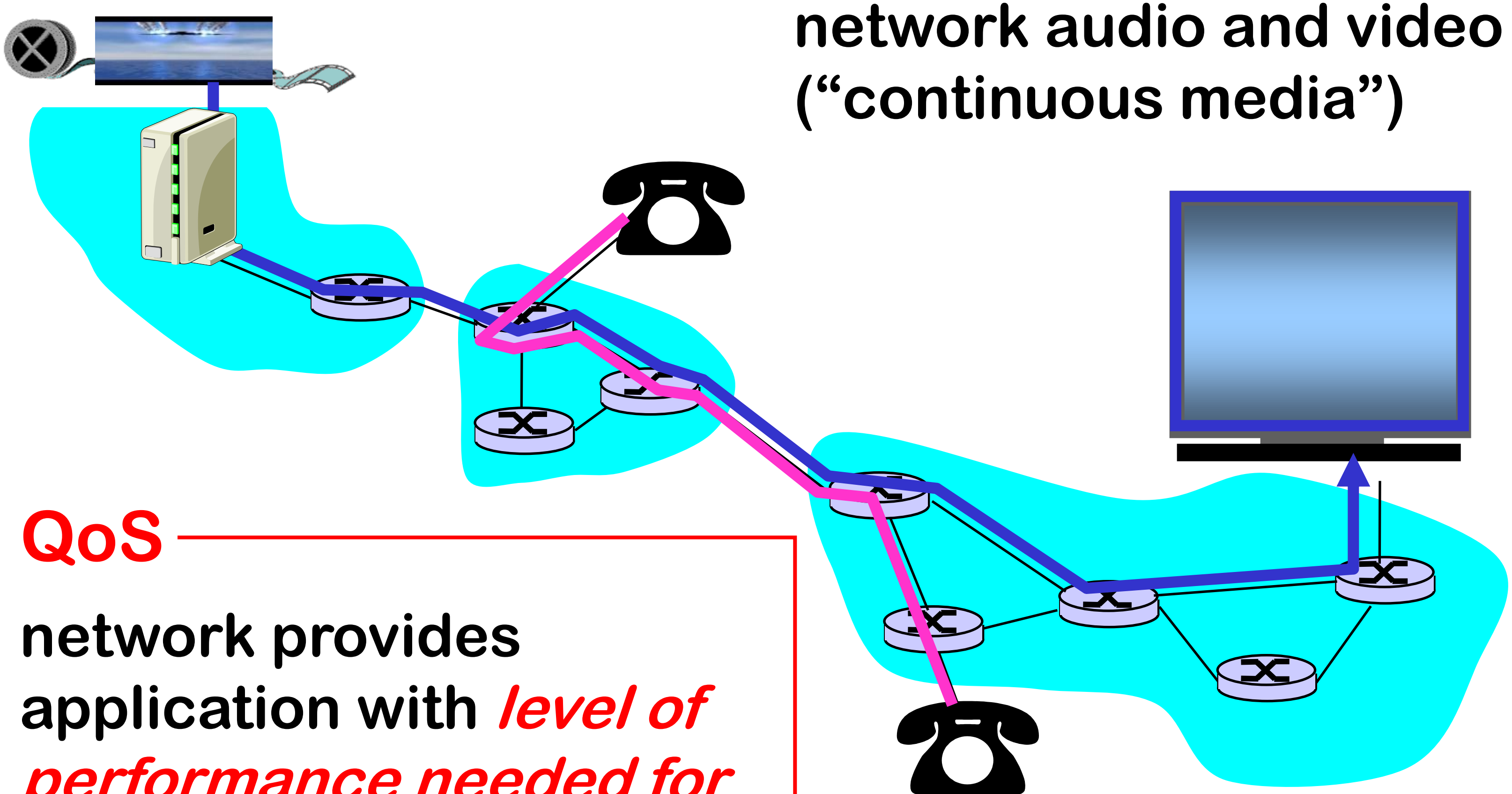
Lecture Overview

- During this lecture, we will
  - Classify emerging multimedia applications
  - Understand Video Streaming, SIP etc.
  - Identify the network services the apps need
  - Learn Mechanisms for providing QoS Network components

- Recommended reading
  - Chapter 19 (Stallings)

Multimedia, Quality of Service: What is it?

Multimedia applications: network audio and video ("continuous media")



QoS

network provides application with *level of performance needed for application to function.*

MM Networking Applications

<u>Classes of MM applications:</u>

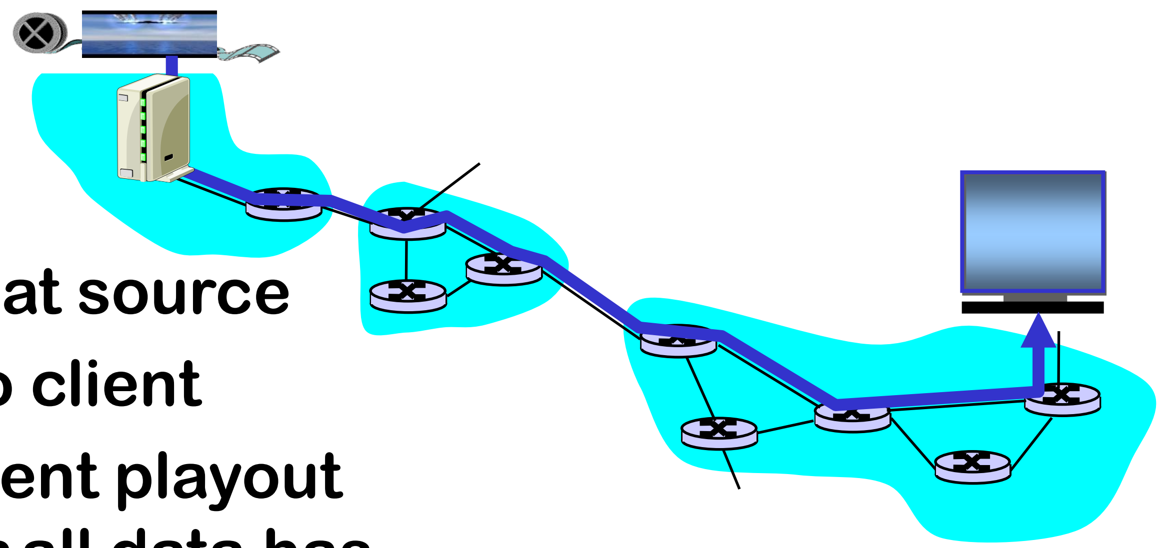
1) Streaming stored audio and video
2) Streaming live audio and video
3) Real-time interactive audio and video

Jitter is the variability of packet delays within the same packet stream

<u>Fundamental characteristics:</u>

- Typically **delay sensitive**
  - end-to-end delay
  - delay jitter
- But **loss tolerant**: infrequent losses cause minor glitches
- Antithesis of data, which are loss intolerant but delay tolerant.

Streaming Stored Multimedia



Streaming:

- media stored at source
- transmitted to client
- streaming: client playout begins *before* all data has arrived
- timing constraint for still-to-be transmitted data: in time for playout

Streaming Live Multimedia

Examples:

- Internet radio talk show
- Live sporting event

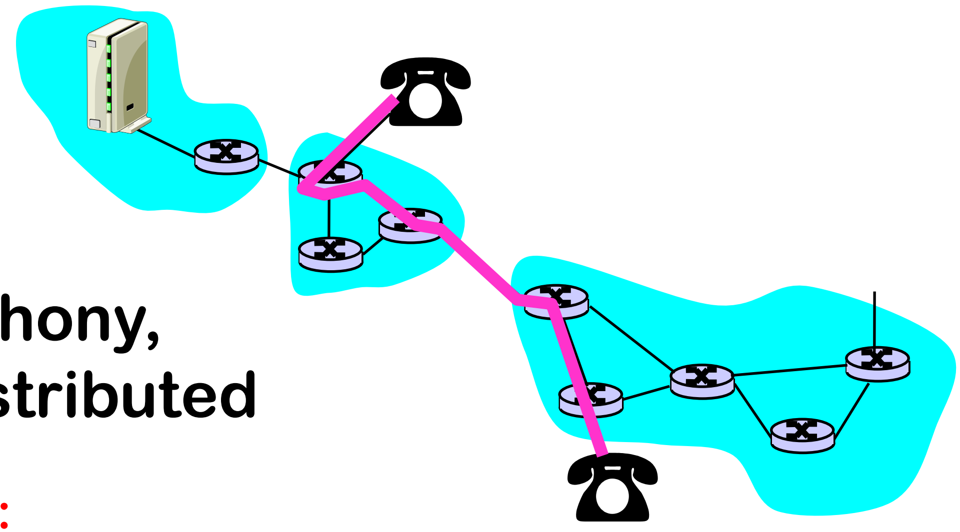
Streaming

- playback buffer
- playback can lag tens of seconds after transmission
- still have timing constraint

Interactivity

- fast forward impossible
- rewind, pause possible!

Interactive, Real-Time Multimedia

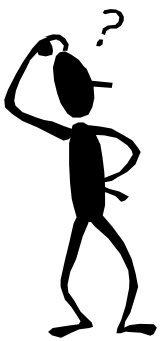


- applications: IP telephony, video conference, distributed interactive worlds
- end-end delay requirements:
  - audio: < 150 msec good, < 400 msec OK
    - ✓ includes application-level (packetization) and network delays
    - ✓ higher delays noticeable, impair interactivity
- session initialization
  - how does callee advertise its IP address, port number, encoding algorithms?

Multimedia Over Today’s Internet

TCP/UDP/IP: “best-effort service”

- *no* guarantees on delay, loss



But you said multimedia apps requires QoS and level of performance to be effective!

Today’s Internet multimedia applications use application-level techniques to mitigate (as best possible) effects of delay, loss

How should the Internet evolve to better support multimedia?

Integrated services philosophy:
- Fundamental changes in Internet so that apps can reserve end-to-end bandwidth
- Requires new, complex software in hosts & routers

Laissez-faire
- no major changes
- more bandwidth when needed
- content distribution, application-layer multicast
  - application layer

Differentiated services philosophy:
- Fewer changes to Internet infrastructure, yet provide 1st and 2nd class service.

What's your opinion?

Streaming Stored Multimedia

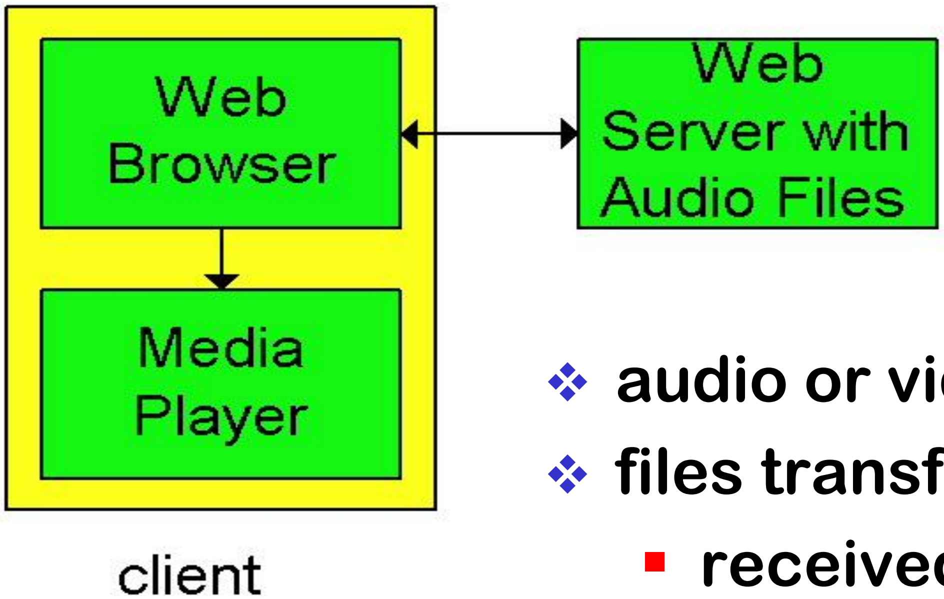
Application-level streaming techniques for making the best out of best effort service:

- client side buffering
- use of UDP versus TCP
- multiple encodings of multimedia

Media Player

- jitter removal
- decompression
- error concealment
- graphical user interface w/ controls for interactivity

Internet multimedia: simplest approach

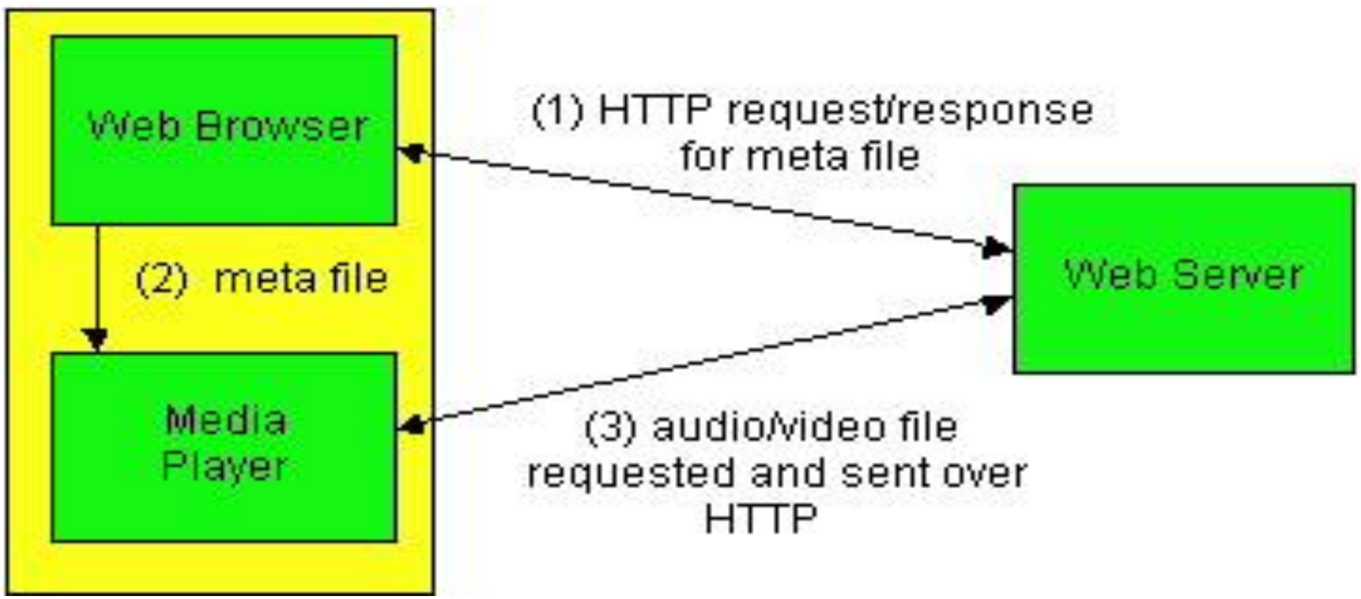


- audio or video stored in file
- files transferred as HTTP object
  - received in entirety at client
  - then passed to player

audio, video not streamed:

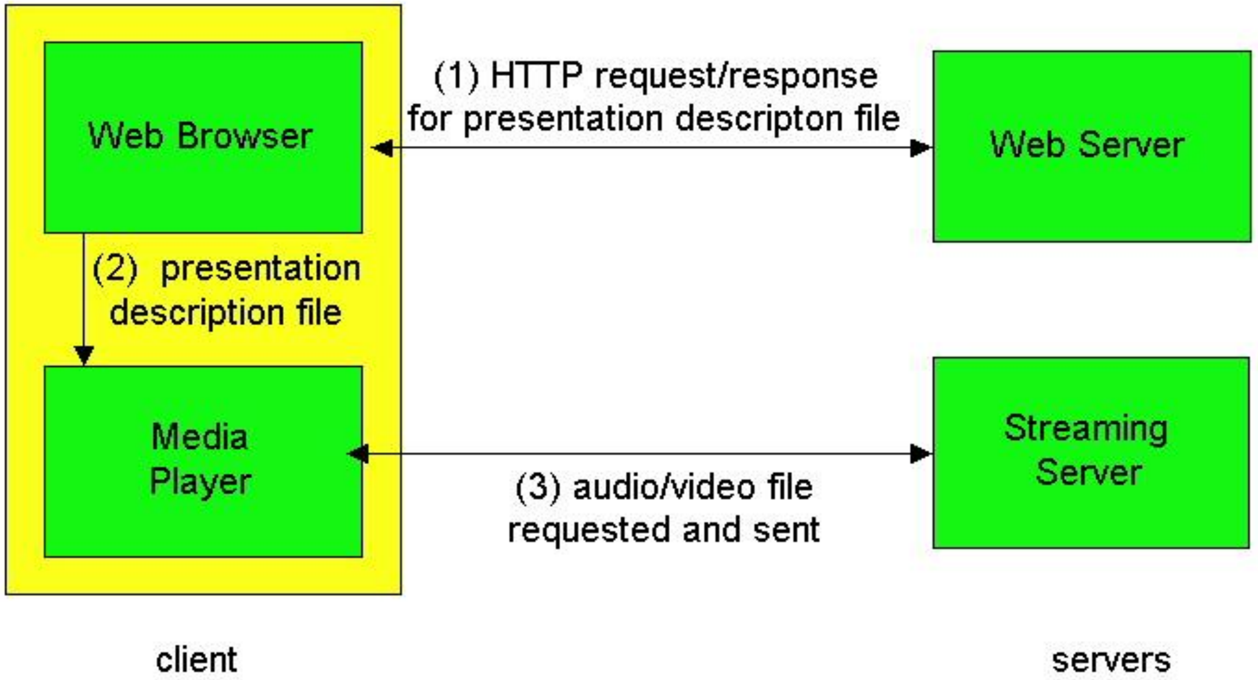
- no, “pipelining,” long delays until playout!

Internet multimedia: streaming approach



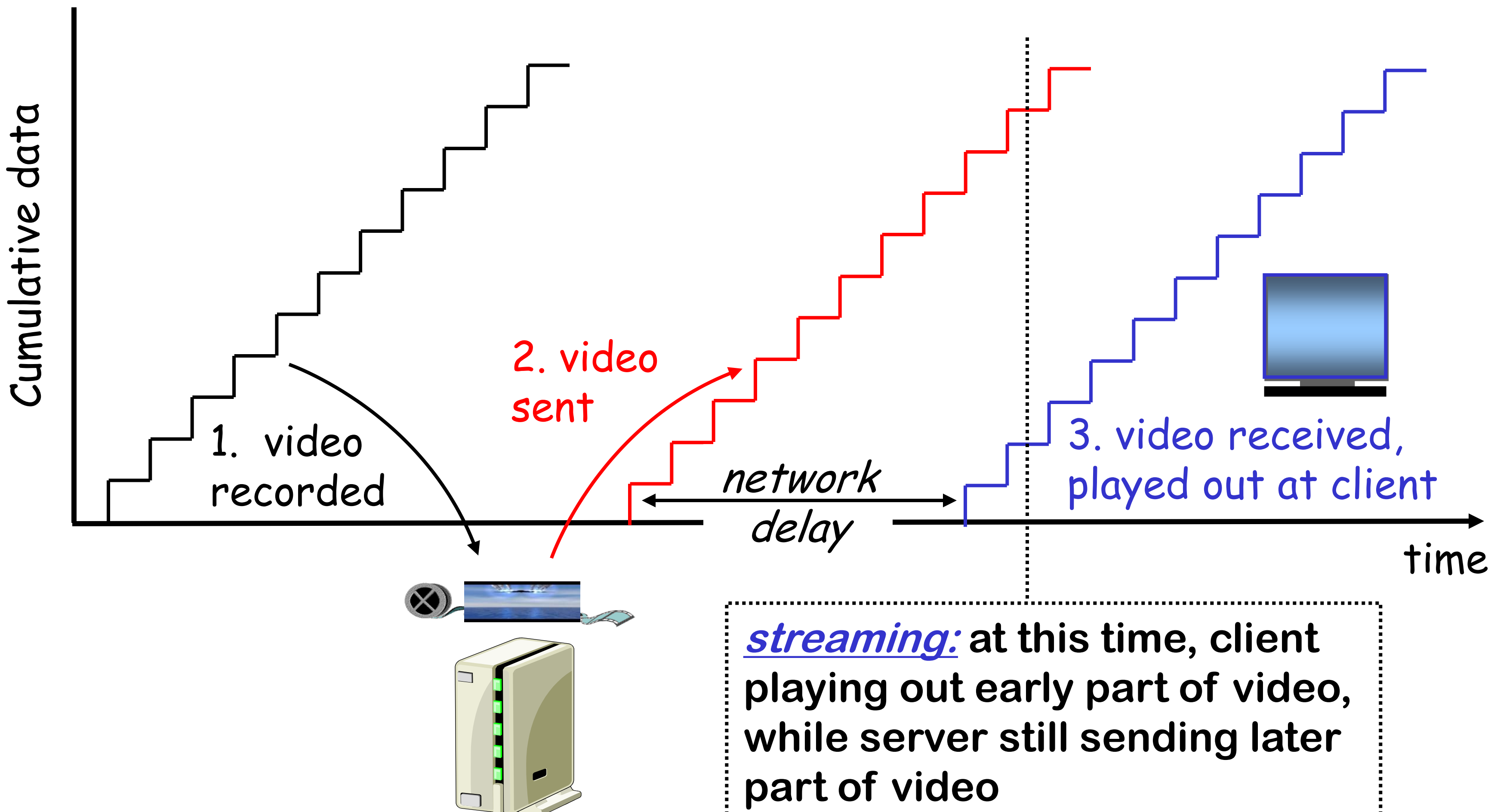
- browser GETs **metafile**
- browser launches player, passing metafile
- player contacts server
- server **streams** audio/video to player

Streaming from a streaming server



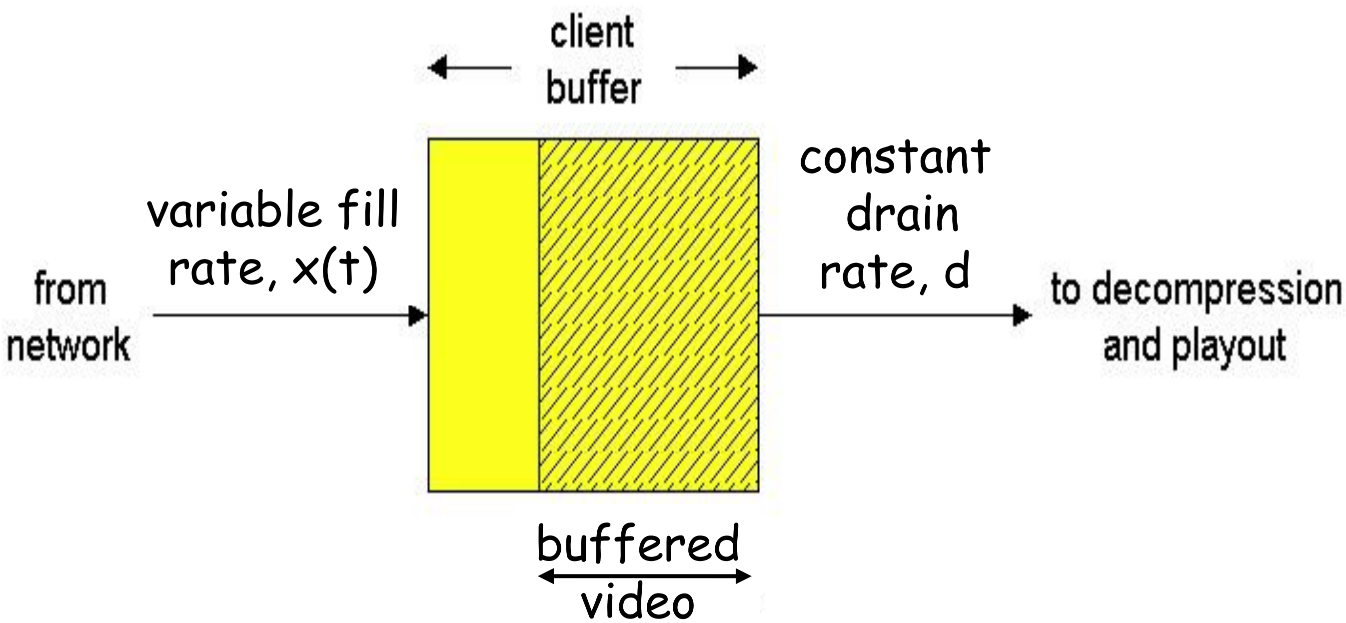
- This architecture allows for non-HTTP protocol between server and media player
- Can also use UDP instead of TCP.

Streaming Stored Multimedia and Buffering



streaming: at this time, client playing out early part of video, while server still sending later part of video

Streaming Multimedia: Client Buffering



- Client-side buffering, playout delay compensate for network-added delay, delay jitter

Streaming Multimedia: UDP or TCP?

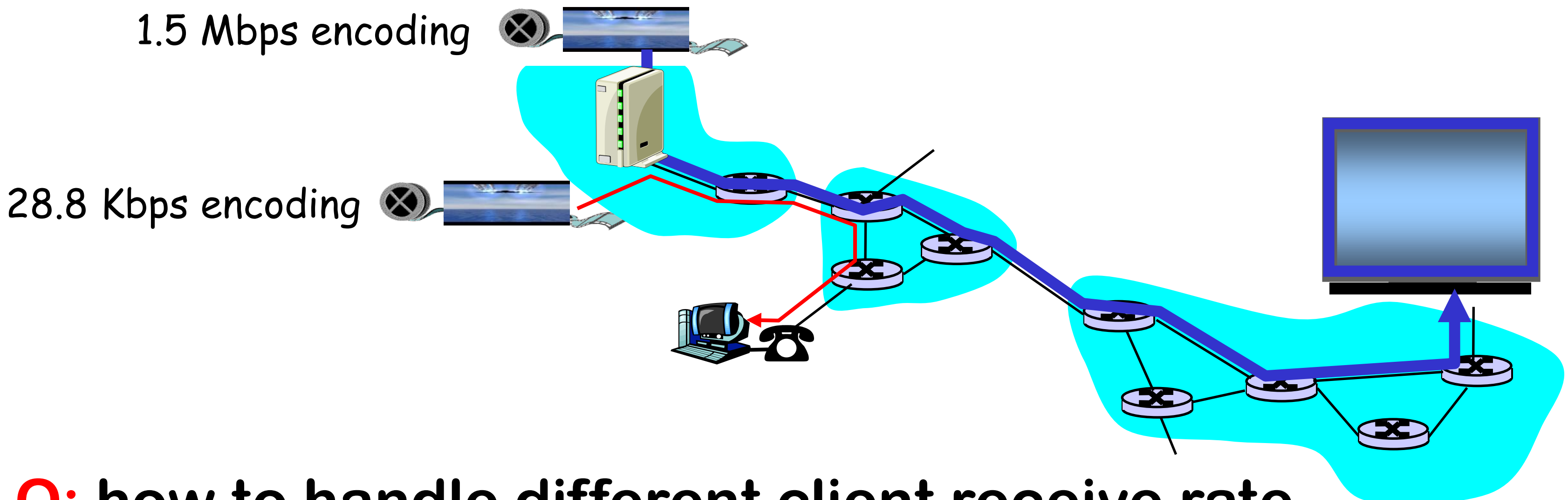
UDP

- server sends at rate appropriate for client (oblivious to network congestion !)
  - often send rate = encoding rate = constant rate
  - then, fill rate = constant rate - packet loss
- short playout delay (2-5 seconds) to compensate for network delay jitter
- error recover: time permitting

TCP

- send at maximum possible rate under TCP
- fill rate fluctuates due to TCP congestion control
- larger playout delay: smooth TCP delivery rate
- HTTP/TCP passes more easily through firewalls

Streaming Multimedia: client rate(s)



Q: how to handle different client receive rate capabilities?

- 28.8 Kbps dialup
- 100Mbps Ethernet

A: server stores, transmits multiple copies of video, encoded at different rates

Real-time interactive applications

- PC-2-PC phone
  - instant messaging services are providing this
  - Skype
- PC-2-phone
  - Dialpad
  - Net2phone
- videoconference with Webcams

Introduce Internet Phone by way of an example

- speaker's audio: alternating talk spurts, silent periods.
  - 64 kbps during talk spurt
- pkts generated only during talk spurts
  - 20 msec chunks at 8 Kbytes/sec: 160 bytes data
- application sends UDP segment into socket every 20 msec during talkspurt.

Internet Phone: Packet Loss and Delay

- network loss: IP datagram lost due to network congestion (router buffer overflow)
- delay loss: IP datagram arrives too late for playout at receiver
  - delays: processing, queueing in network; end-system (sender, receiver) delays
  - typical maximum tolerable delay: 400 ms
- loss tolerance: depending on voice encoding, losses concealed, packet loss rates between 1% and 10% can be tolerated.

- Consider the end-to-end delays of two consecutive packets: difference can be more or less than 20 msec

Summary: Internet Multimedia: bag of tricks

- use UDP to avoid TCP congestion control (delays) for time-sensitive traffic
- client-side adaptive playout delay: to compensate for delay
- server side matches stream bandwidth to available client-to-server path bandwidth
  - chose among pre-encoded stream rates
  - dynamic server encoding rate
- error recovery (on top of UDP)
  - FEC, interleaving
  - retransmissions, time permitting
  - conceal errors: repeat nearby data

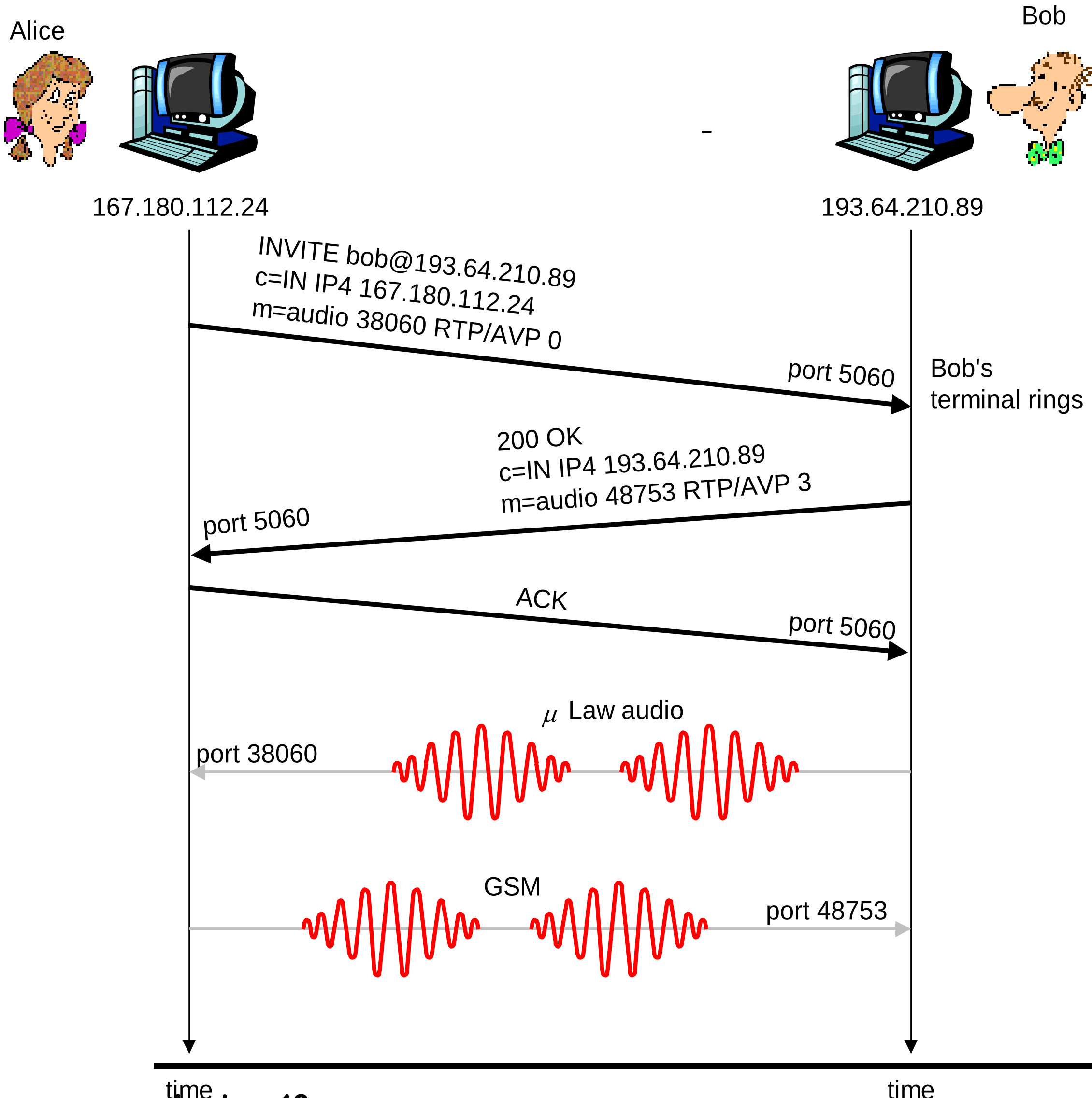
Session Initiation Protocol (SIP)

- Session Initiation Protocol
- Comes from IETF

SIP long-term vision

- All telephone calls and video conference calls take place over the Internet
- People are identified by names or e-mail addresses, rather than by phone numbers.
- You can reach the callee, no matter where the callee roams, no matter what IP device the callee is currently using.

Setting up a call to a known IP address



- Alice's SIP invite message indicates her port number & IP address. Indicates encoding that Alice prefers to receive (PCM ulaw)
- Bob's 200 OK message indicates his port number, IP address & preferred encoding (GSM)
- SIP messages can be sent over TCP or UDP; here sent over RTP/UDP.
- Default SIP port number is 5060.

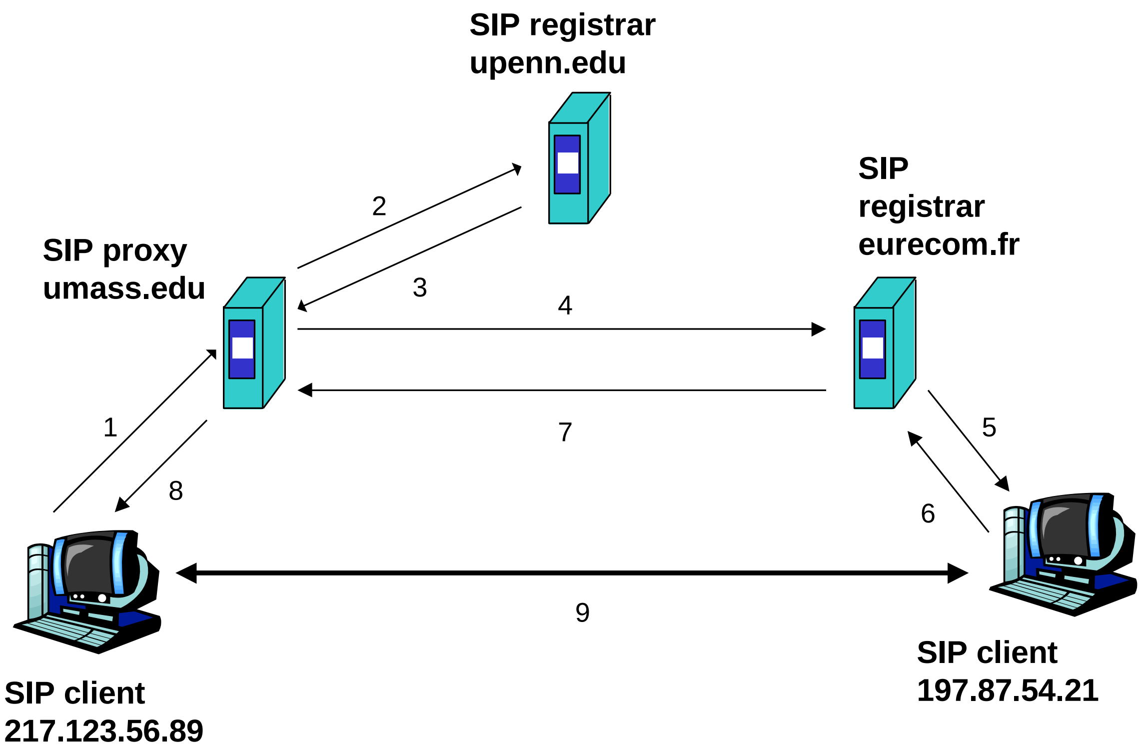
Name translation and user locataion

- Caller wants to call callee, but only has callee's name or e-mail address.
- Need to get IP address of callee's current host:
  - user moves around
  - DHCP protocol
  - user has different IP devices (PC, PDA, car device)
- Result can be based on:
  - time of day (work, home)
  - caller (don't want boss to call you at home)
  - status of callee (calls sent to voicemail when callee is already talking to someone)

Example

Caller jim@umass.edu
with places a
call to keith@upenn.edu

(1) Jim sends INVITE message to umass SIP proxy. (2) Proxy forwards request to upenn registrar server.
(3) upenn server returns redirect response, indicating that it should try keith@eurecom.fr



(4) umass proxy sends INVITE to eurecom registrar. (5) eurecom regristrar forwards INVITE to 197.87.54.21, which is running keith's SIP client. (6-8) SIP response sent back (9) media sent directly between clients.
Note: also a SIP ack message, which is not shown.

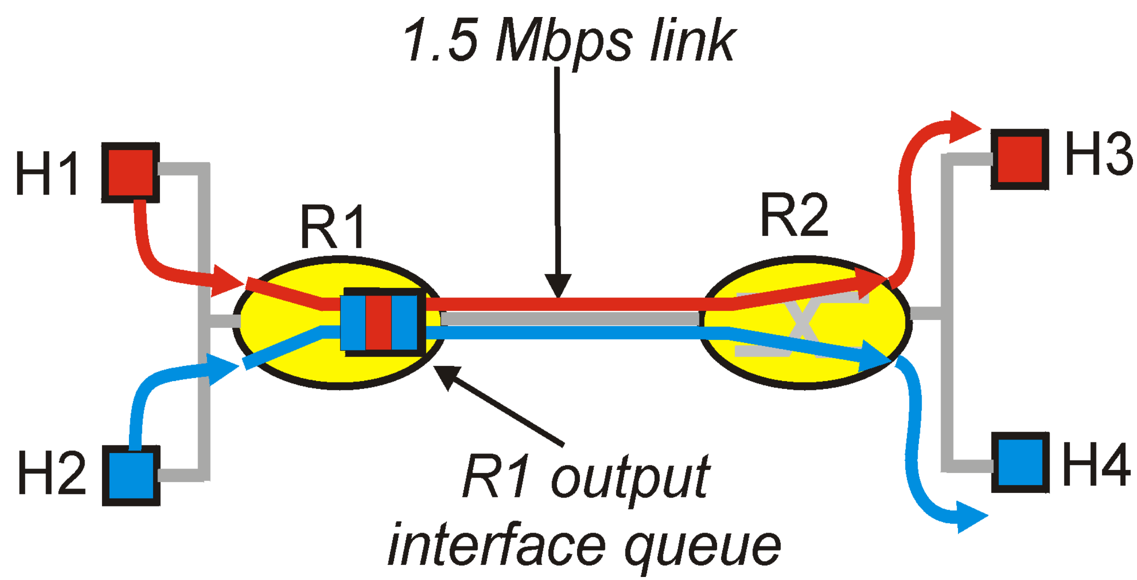
Improving QOS in IP Networks

Thus far: "making the best of best effort"

Future: next generation Internet with QoS guarantees

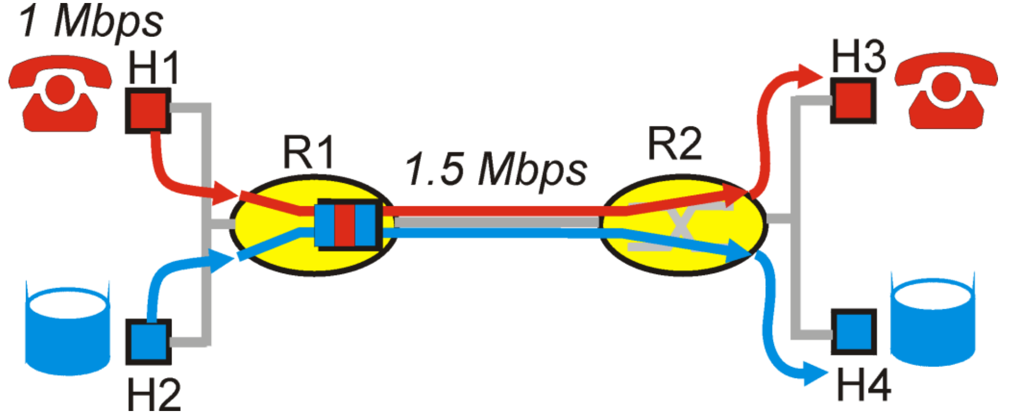
- **RSVP:** signaling for resource reservations
- **Differentiated Services:** differential guarantees
- **Integrated Services:** firm guarantees

- simple model for sharing and congestion studies:



Principles for QOS Guarantees

- Example: 1Mbps I P phone, FTP share 1.5 Mbps link.
  - bursts of FTP can congest router, cause audio loss
  - want to give priority to audio over FTP

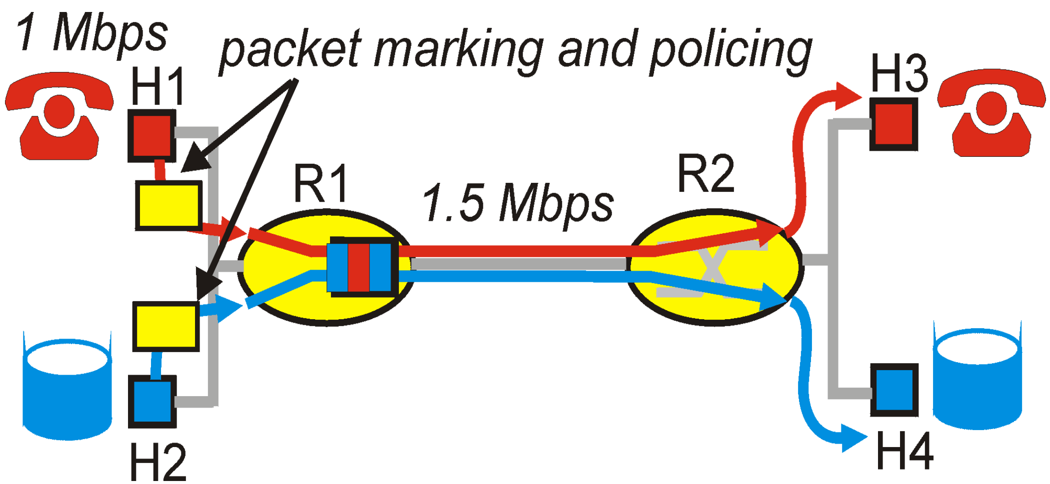


Principle 1

packet marking needed for router to distinguish between different classes; and new router policy to treat packets accordingly

Principles for QOS Guarantees (more)

- what if applications misbehave (audio sends higher than declared rate)
  - policing: force source adherence to bandwidth allocations
- marking and policing at network edge:
  - similar to ATM UNI (User Network Interface)

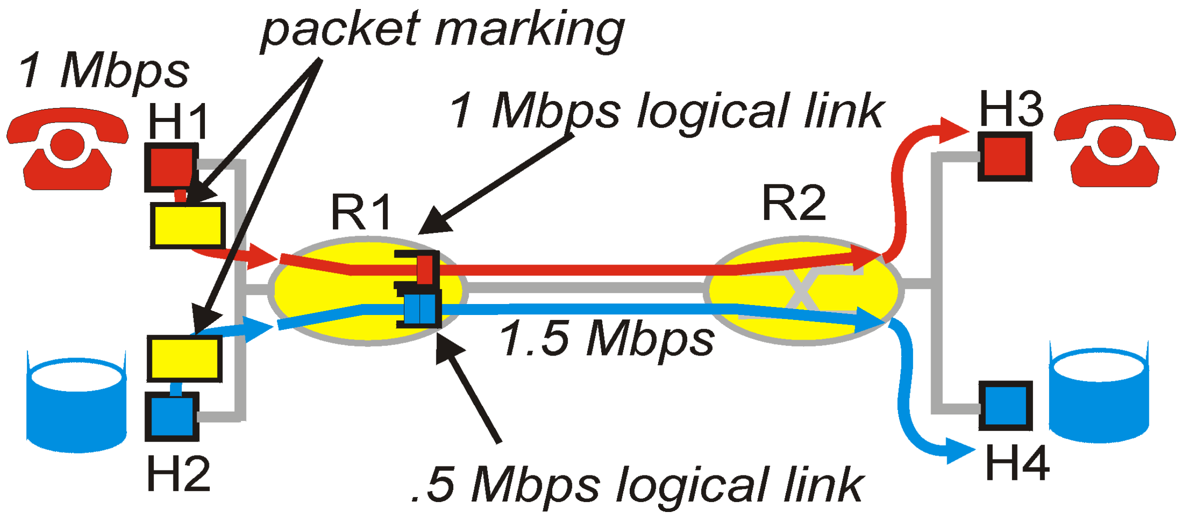


Principle 2

provide protection (*isolation*) for one class from others

Principles for QOS Guarantees (more)

- Allocating *fixed* (non-sharable) bandwidth to flow: *inefficient* use of bandwidth if flows doesn't use its allocation

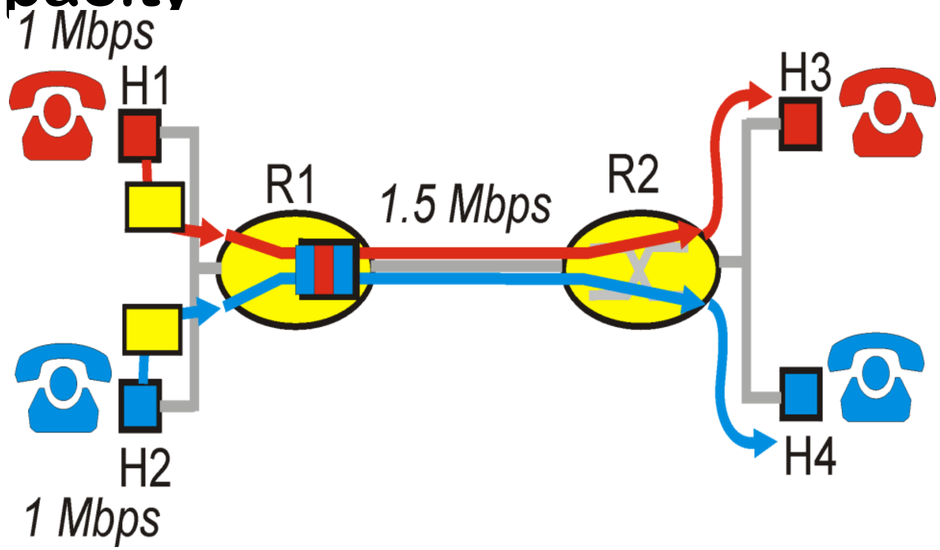


Principle 3

While providing isolation, it is desirable to use resources as efficiently as possible

Principles for QOS Guarantees (more)

- *Basic fact of life:* can not support traffic demands beyond link capacity

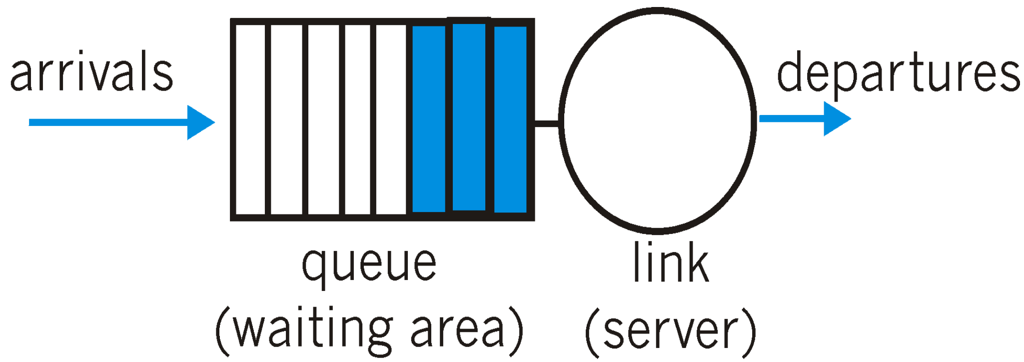


Principle 4

Call Admission: flow declares its needs, network may block call (e.g., busy signal) if it cannot meet needs

Scheduling And Policing Mechanisms

- **scheduling:** choose next packet to send on link
- **FIFO (first in first out) scheduling:** send in order of arrival to queue
  - real-world example?
  - **discard policy:** if packet arrives to full queue: who to discard?
    - ✓ Tail drop: drop arriving packet
    - ✓ priority: drop/remove on priority basis
    - ✓ random: drop/remove randomly



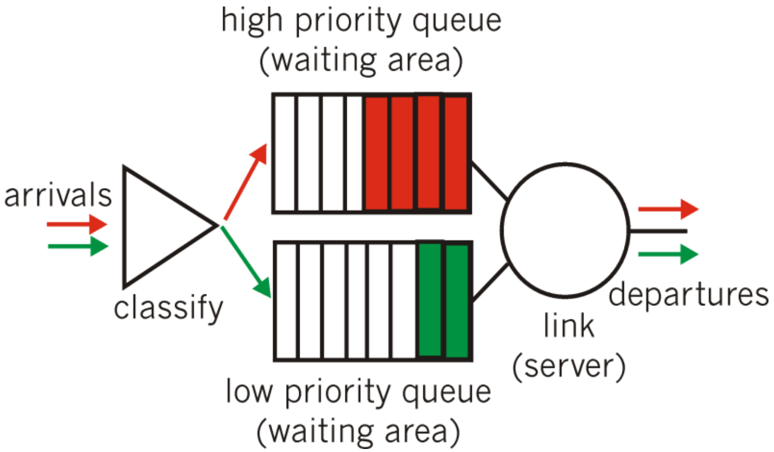
Scheduling Policies: more

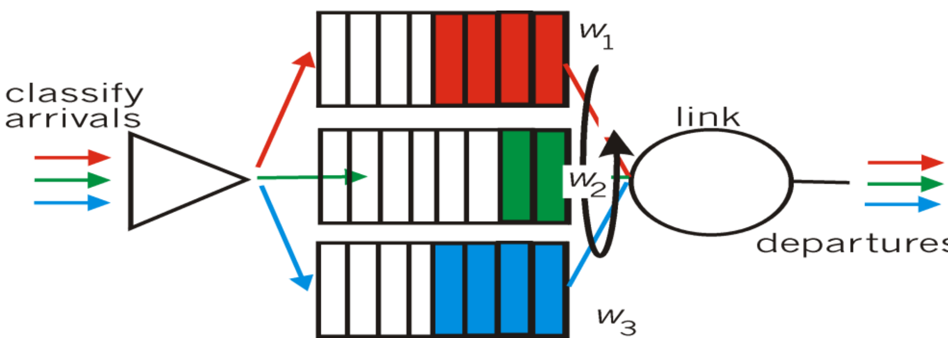
Priority scheduling: transmit highest priority queued packet

- multiple *classes*, with different priorities
  - class may depend on marking or other header info, e.g. IP source/dest, port numbers, etc..

Weighted Fair Queuing:

- generalized Round Robin
- each class gets weighted amount of service in each cycle





Policing Mechanisms

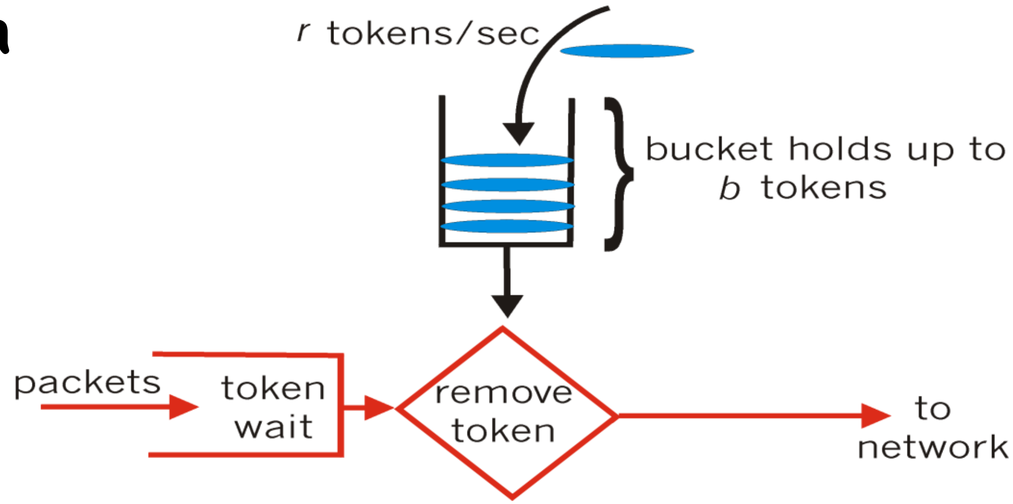
Goal: limit traffic to not exceed declared parameters

Three common-used criteria:

- *(Long term) Average Rate:* how many pkts can be sent per unit time (in the long run)
  - crucial question: what is the interval length: 100 packets per sec or 6000 packets per min have same average!
- *Peak Rate:* e.g., 6000 pkts per min. (ppm) avg.; 1500 ppm peak rate
- *(Max.) Burst Size:* max. number of pkts sent consecutively (with no intervening idle)

Policing Mechanisms

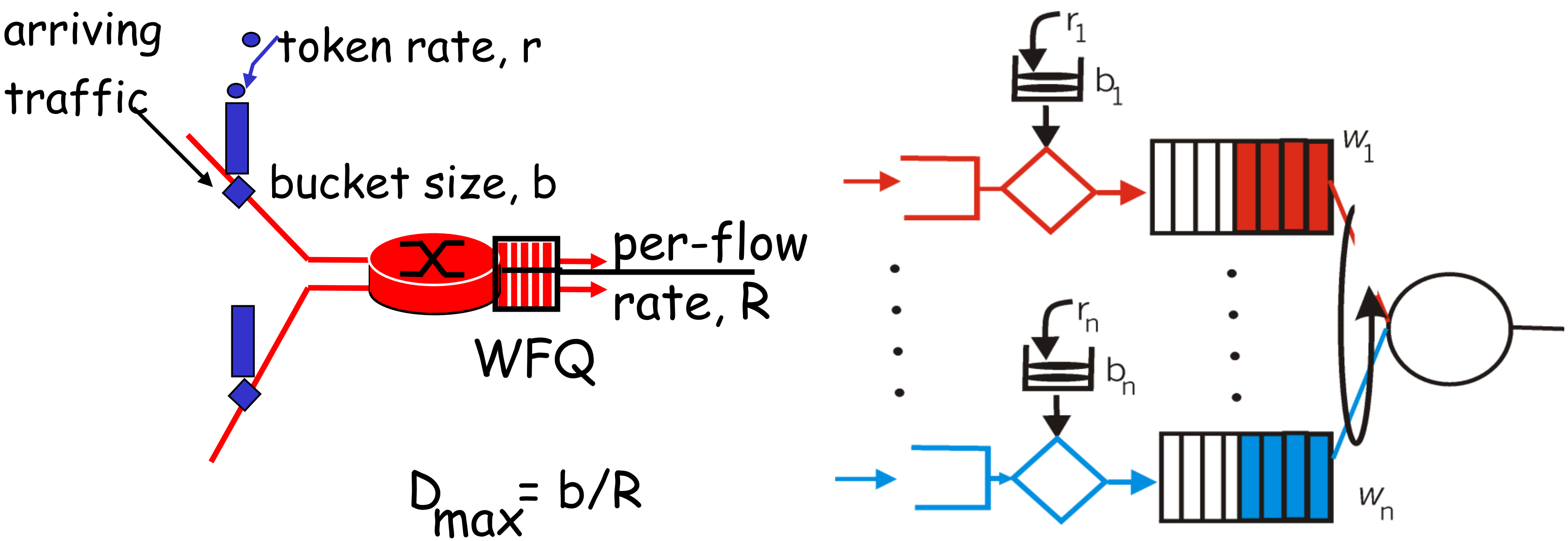
Token Bucket: limit input to specified Burst Size and Average Ra



- bucket can hold b tokens
- tokens generated at rate *r token/sec* unless bucket full
- *over interval of length t: number of packets admitted less than or equal to (r t + b).*

Policing Mechanisms (more)

- token bucket, WFQ combine to provide guaranteed upper bound on delay, i.e., ***QoS guarantee***!



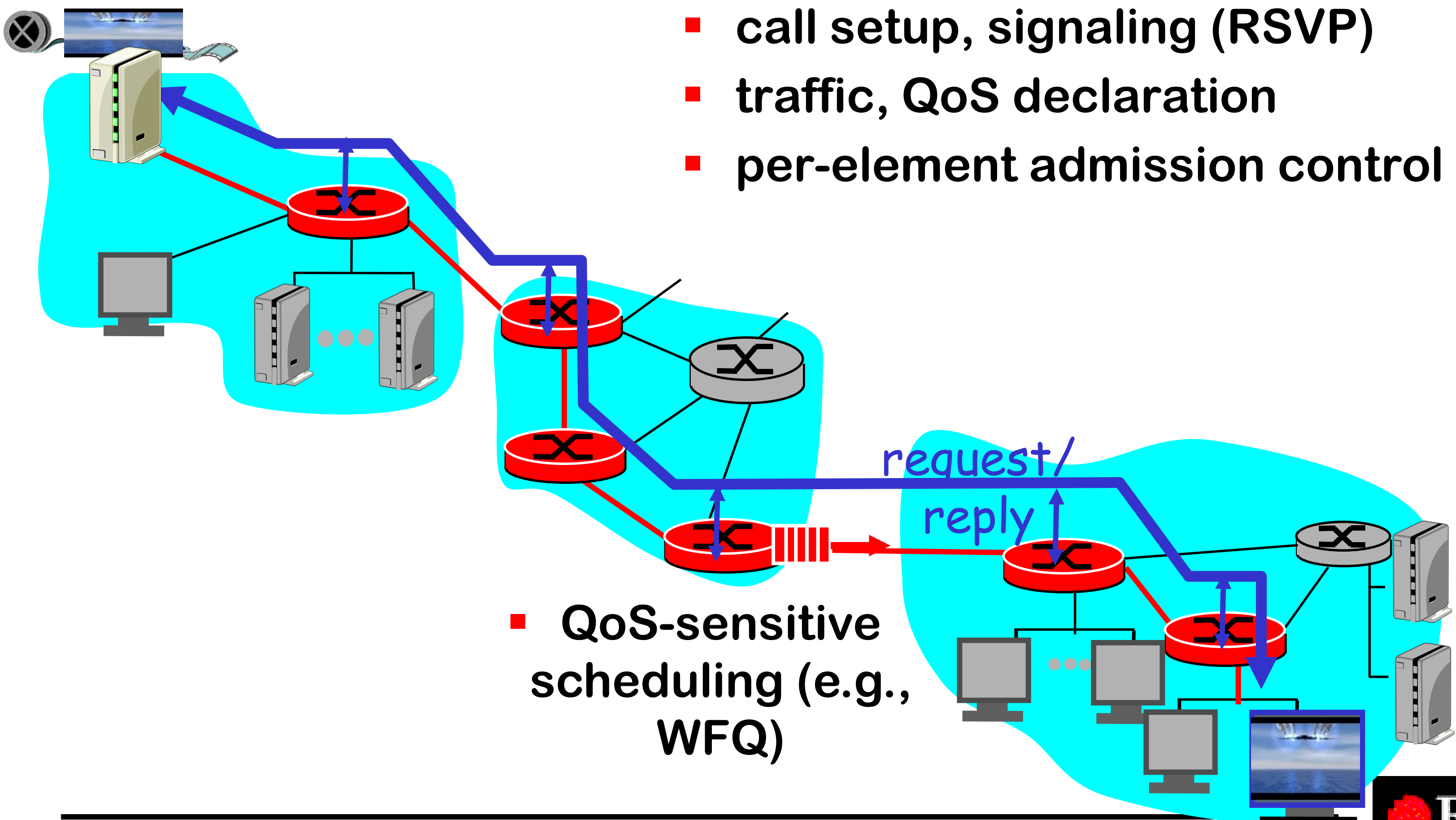
IETF Integrated Services

- architecture for providing QOS guarantees in IP networks for individual application sessions
- resource reservation: routers maintain state info (a la VC) of allocated resources, QoS req's
- admit/deny new call setup requests:

Question: can newly arriving flow be admitted with performance guarantees while not violated QoS guarantees made to already admitted flows?

Intserv: QoS guarantee scenario

- Resource reservation
  - call setup, signaling (RSVP)
  - traffic, QoS declaration
  - per-element admission control



- QoS-sensitive scheduling (e.g., WFQ)

IETF Differentiated Services

Concerns with Intserv:

- **Scalability:** signaling, maintaining per-flow router state difficult with large number of flows
- **Flexible Service Models:** Intserv has only two classes. Also want "qualitative" service classes
  - "behaves like a wire"
  - relative service distinction: Platinum, Gold, Silver

Diffserv approach:

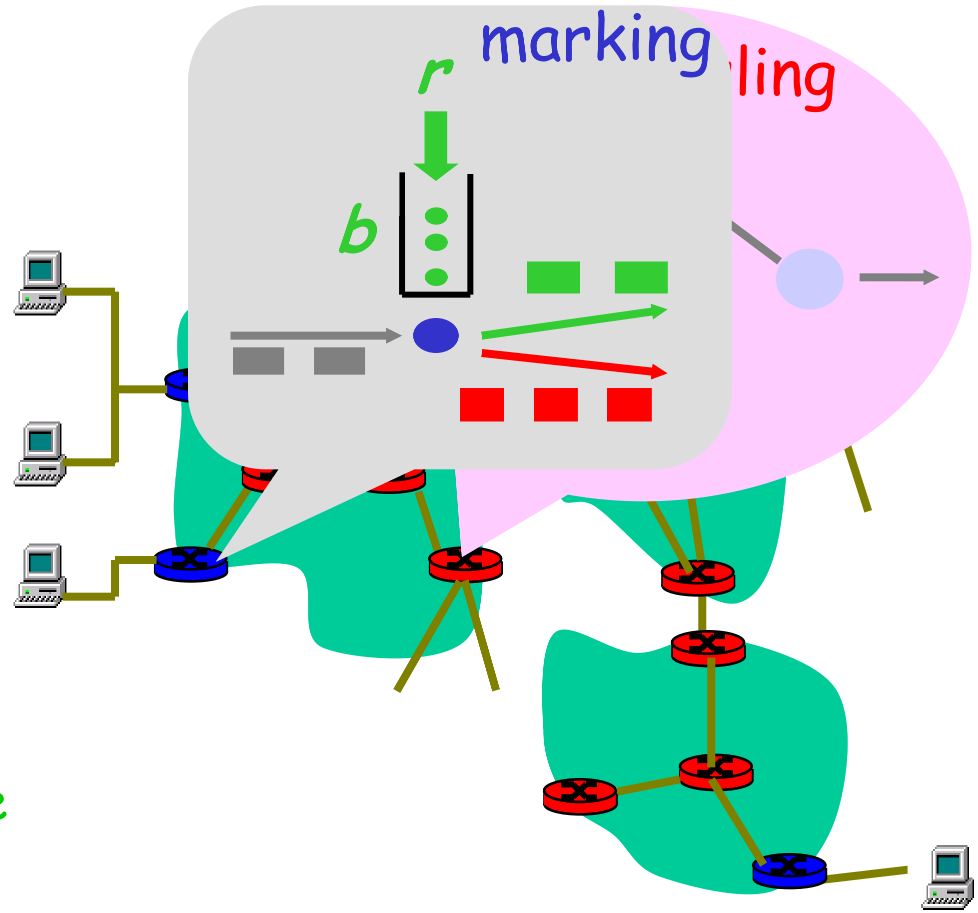
- simple functions in network core, relatively complex functions at edge routers (or hosts)
- Do't define define service classes, provide functional components to build service classes

Diffserv Architecture

Edge router:

- per-flow traffic management

- marks packets as in-profile and out-profile



Core router:

- per class traffic management

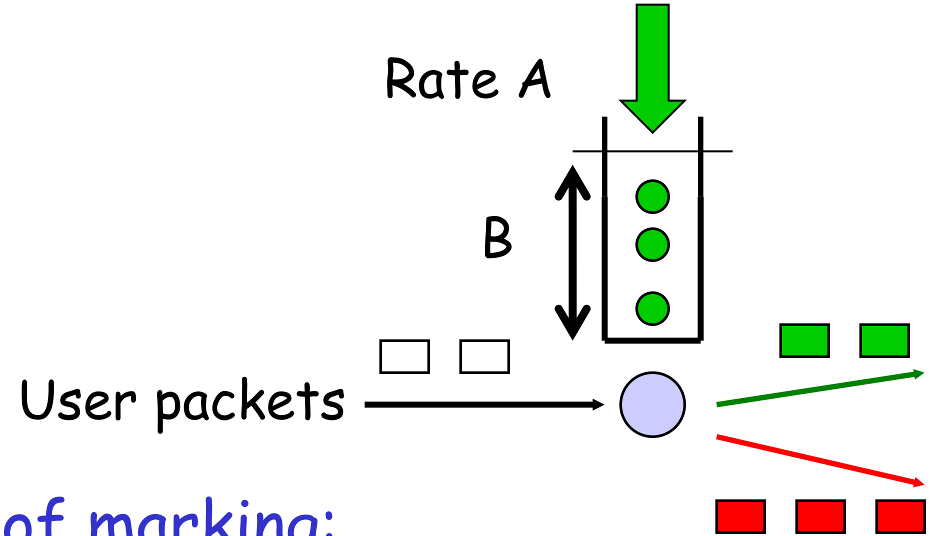
- buffering and scheduling based on marking at edge

- preference given to in-profile packets

- Assured Forwarding

Edge-router Packet Marking

- profile: pre-negotiated rate A, bucket size B
- packet marking at edge based on per-flow profile

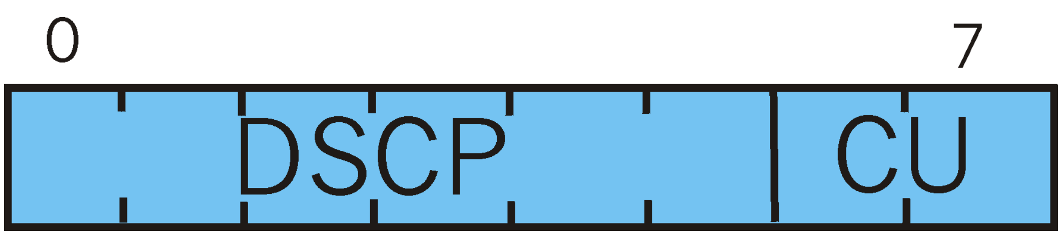


Possible usage of marking:

- class-based marking: packets of different classes marked differently
- intra-class marking: conforming portion of flow marked differently than non-conforming one

Classification and Conditioning

- Packet is marked in the Type of Service (TOS) in IPv4, and Traffic Class in IPv6
- 6 bits used for Differentiated Service Code Point (DSCP) and determine PHB that the packet will receive
- 2 bits are currently unused



Forwarding (PHB)

PHBs being developed:

- **Expedited Forwarding:** pkt departure rate of a class equals or exceeds specified rate
  - logical link with a minimum guaranteed rate
- **Assured Forwarding:** 4 classes of traffic
  - each guaranteed minimum amount of bandwidth
  - each with three drop preference partitions

Multimedia Networking: Summary

- multimedia applications and requirements
- making the best of today's best effort service
- scheduling and policing mechanisms
- next generation Internet: Intserv, RSVP, Diffserv

Summary

- In this lecture, we have understood:
  - Emerging multimedia applications like video streaming, SIP etc.
  - Network services the apps need
  - Mechanisms for providing QoS Network components

Next Time

Good Luck ! Happy Exam !